Discovering patterns of activity in unstructured incident reports at scale

Bronwyn Woods*, Sam Perl

** blwoods@cert.org*

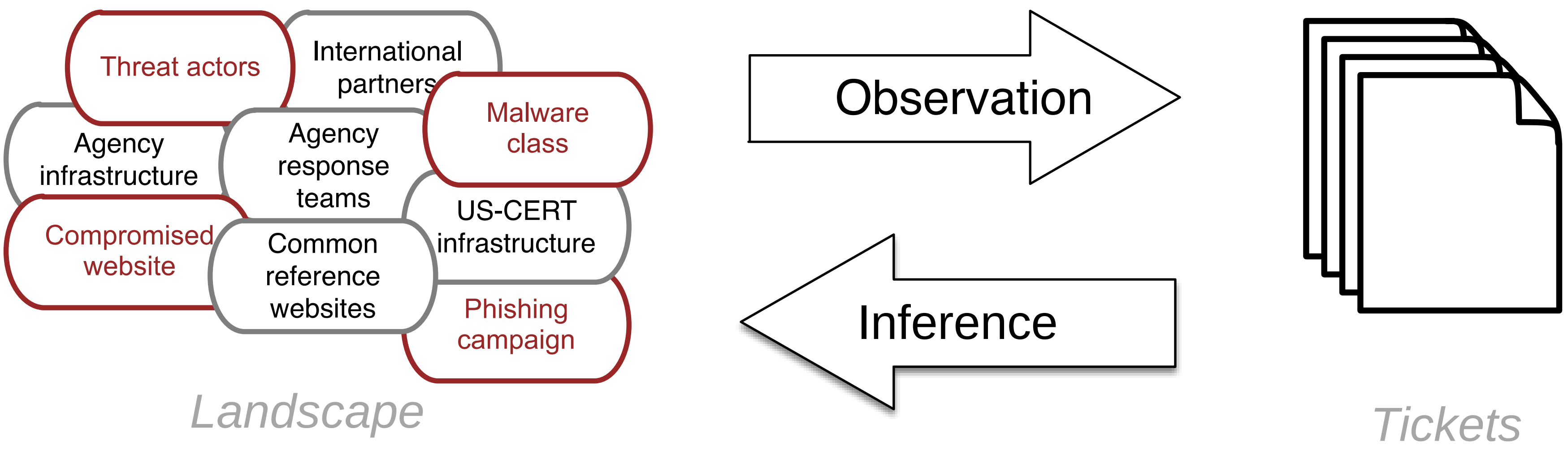
Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

This material is based upon work funded and supported by Department of Homeland Security under Contract No. FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center sponsored by the United States Department of Defense.

DM-0002418

Goal: From tickets to cyber landscape

- US-CERT receives incident reports from a diverse constituency.
- Each ticket is an observation of problematic activity by a particular reporter.
- Taken en masse, we use the tickets as as a statistical sample of observations to learn about the threat and defense landscape.
- Specifically, we infer similarity relationships and functional clusters of indicators using information about reporting patterns.



Some approaches

This talk

Extract indicators and exploit reporting patterns across agencies and tickets.

- Indicator similarity
- Indicator communities

Ask us

Parse free text descriptions of incidents for tagging, topic modeling, and information extraction.

- Exploit regularities in the format of tickets from individual reporters
- Infer and extract frequently reported information
  e.g. cost of incident, resolution status, impact
- More value in tickets without extra cost to reporters.

Data Description

This dataset consists of incident tickets from 2013. Each ticket has:

Structured Fields:

- Reporter information
- Category, subcategory
- Date of submission
- Information about US-CERT ticket processing: assigned group, closure status

Unstructured Field:

- Notes (free text allowed)

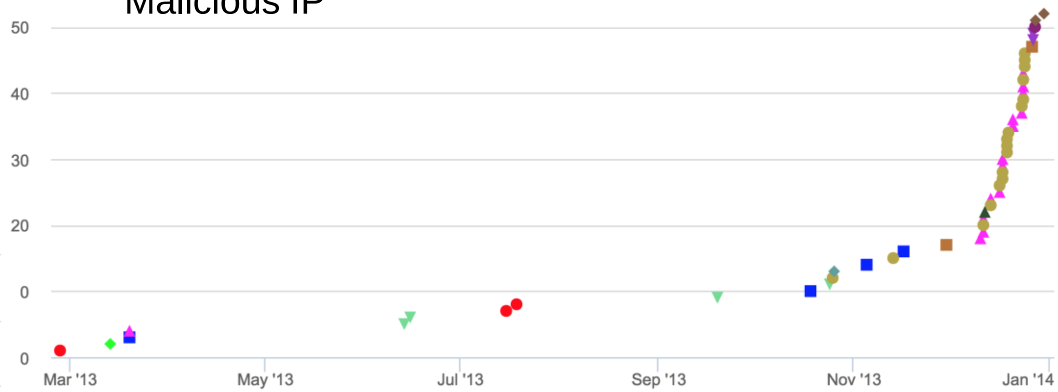
The unstructured notes field contains most of the information about each ticket.

Indicators across tickets

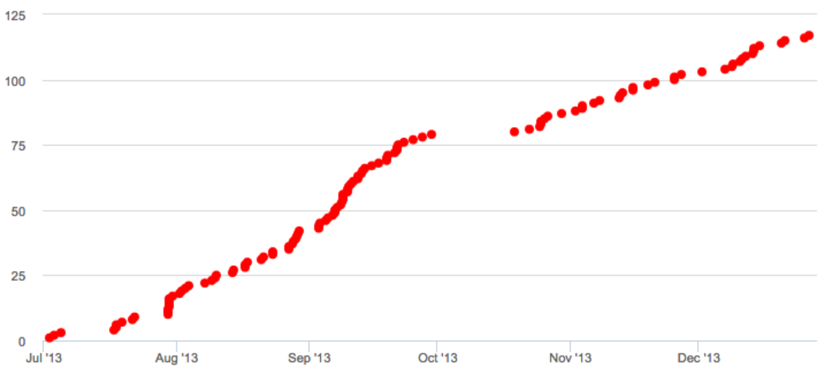
Indicators occur with diverse patterns across tickets, reporters and time.

Time on x axis, count on y axis, color coded by reporter.

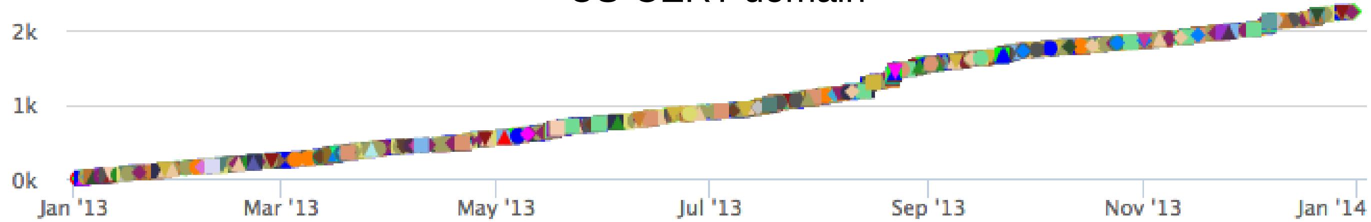
Malicious IP



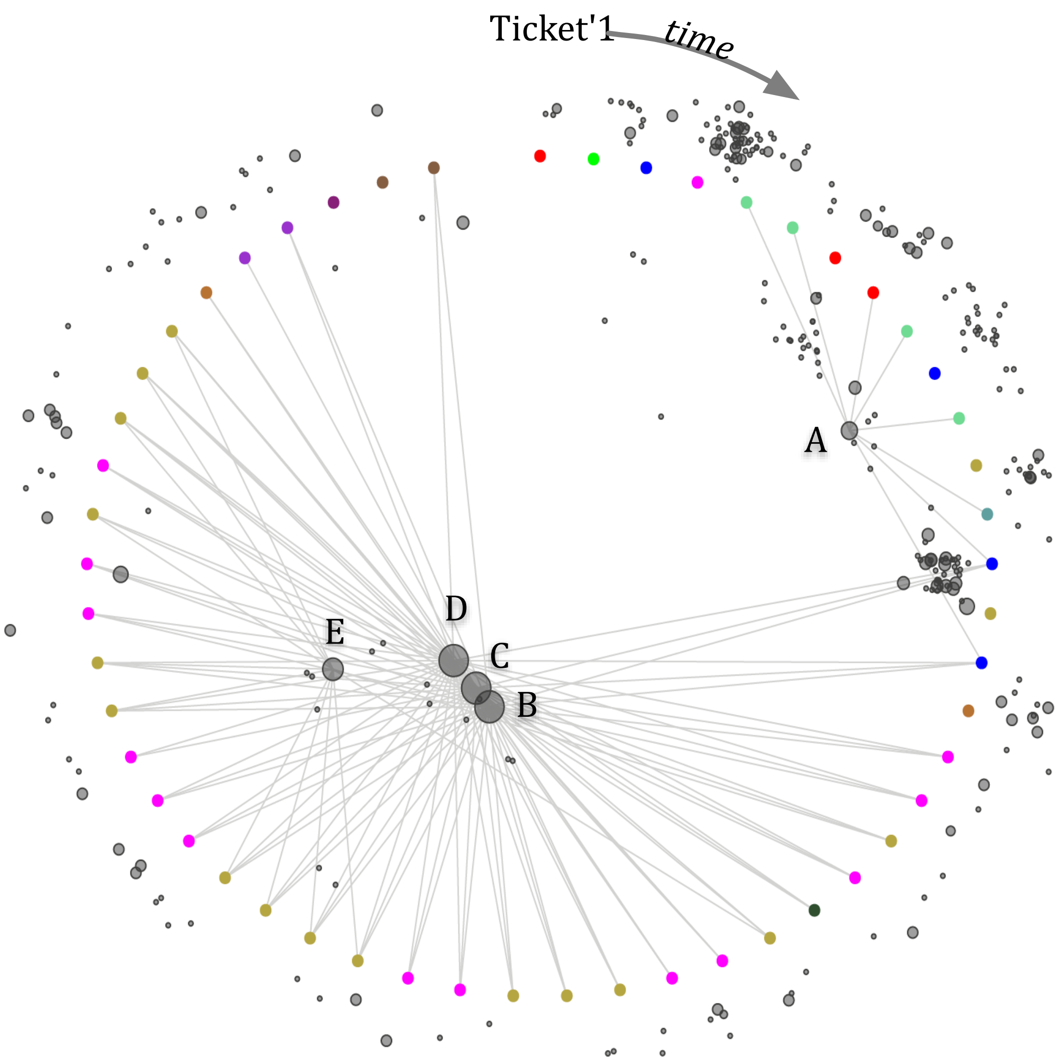
Agency IP



US-CERT domain



Similarity of indicators

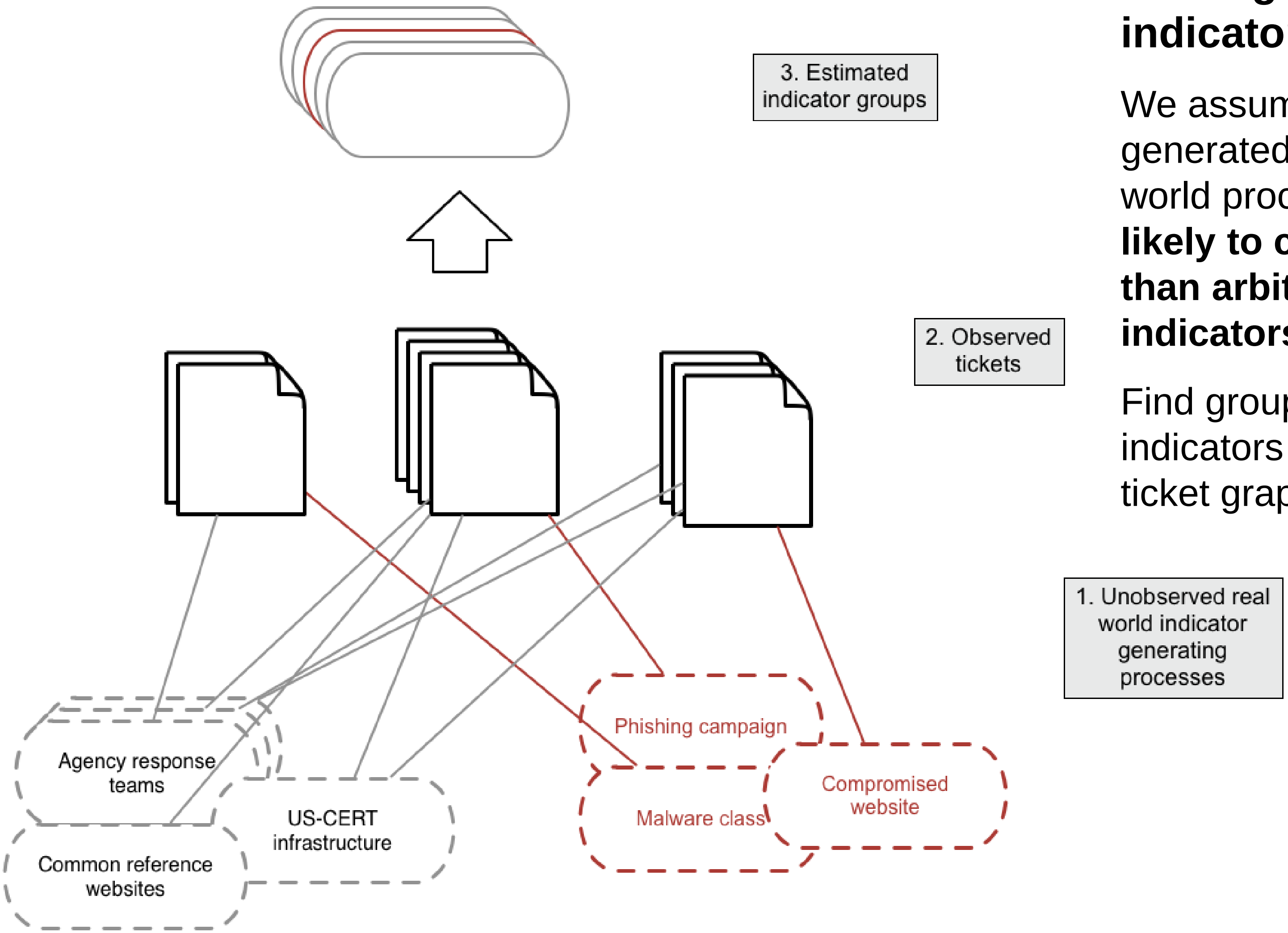


Beginning with a reference indicator, we find indicators similar to it.

Example: a malicious IP

- Colored circles are tickets
- Grey circles are indicators
- Large indicators near center of circle have similar occurrence patterns to the reference indicator.

Indicator communities

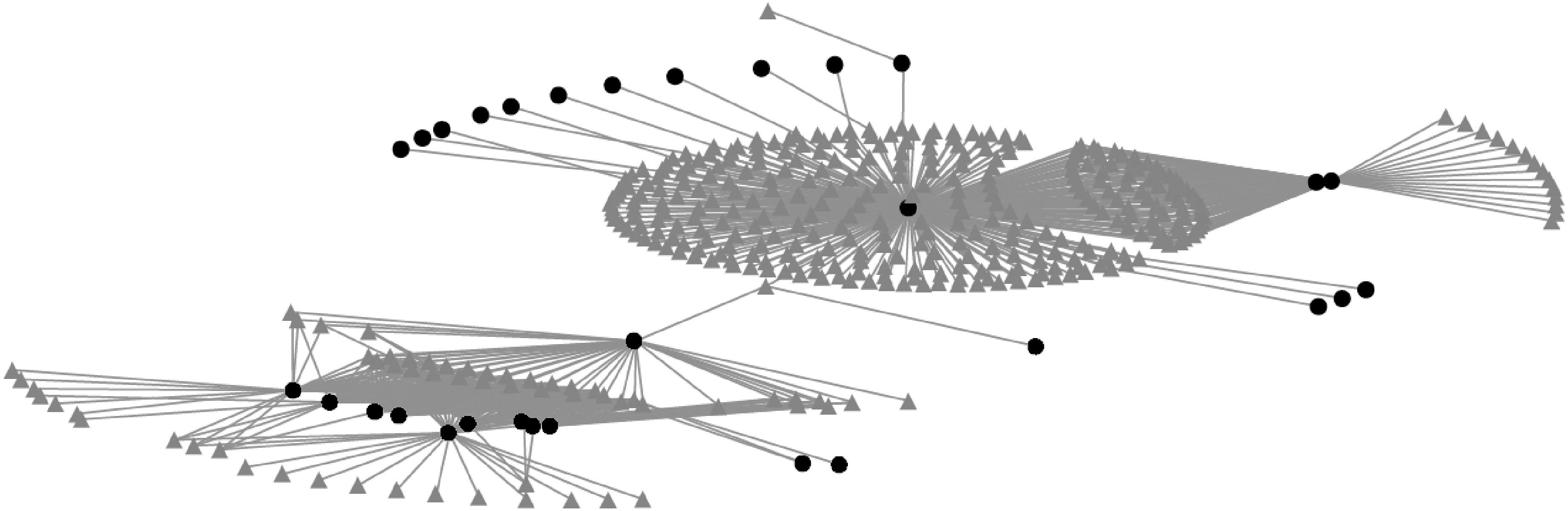


But what if we aren't starting with a reference indicator?

We assume that indicators generated by a coherent real world process will be **more likely to co-occur in tickets than arbitrary pairs of indicators**.

Find groups of highly similar indicators in complete indicator-ticket graph.

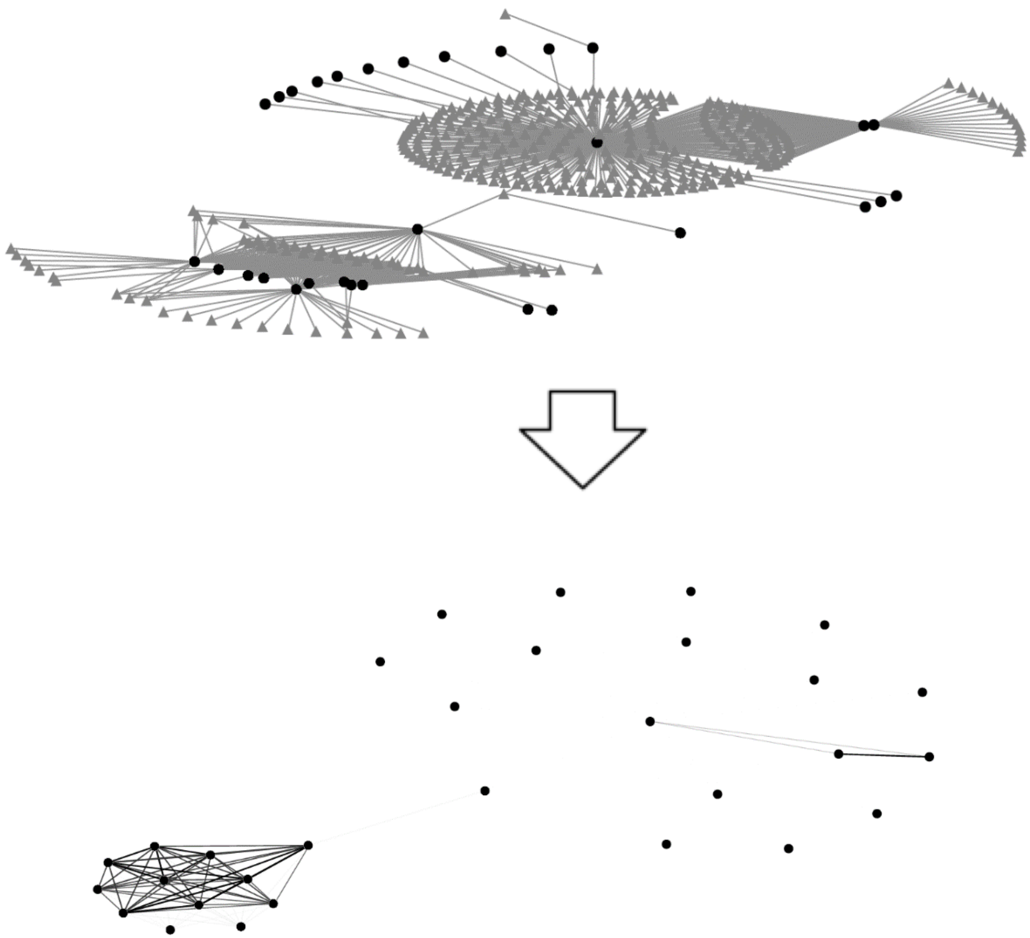
Indicator-ticket graph



A subset of the ticket-indicator graph
(for a small set of selected indicators)

- Tickets are grey triangles
- Indicators are black circles
- Edges connect tickets to the indicators they contain

Indicator-indicator weighted graph



Tickets are observations, focus on relationships between indicators.

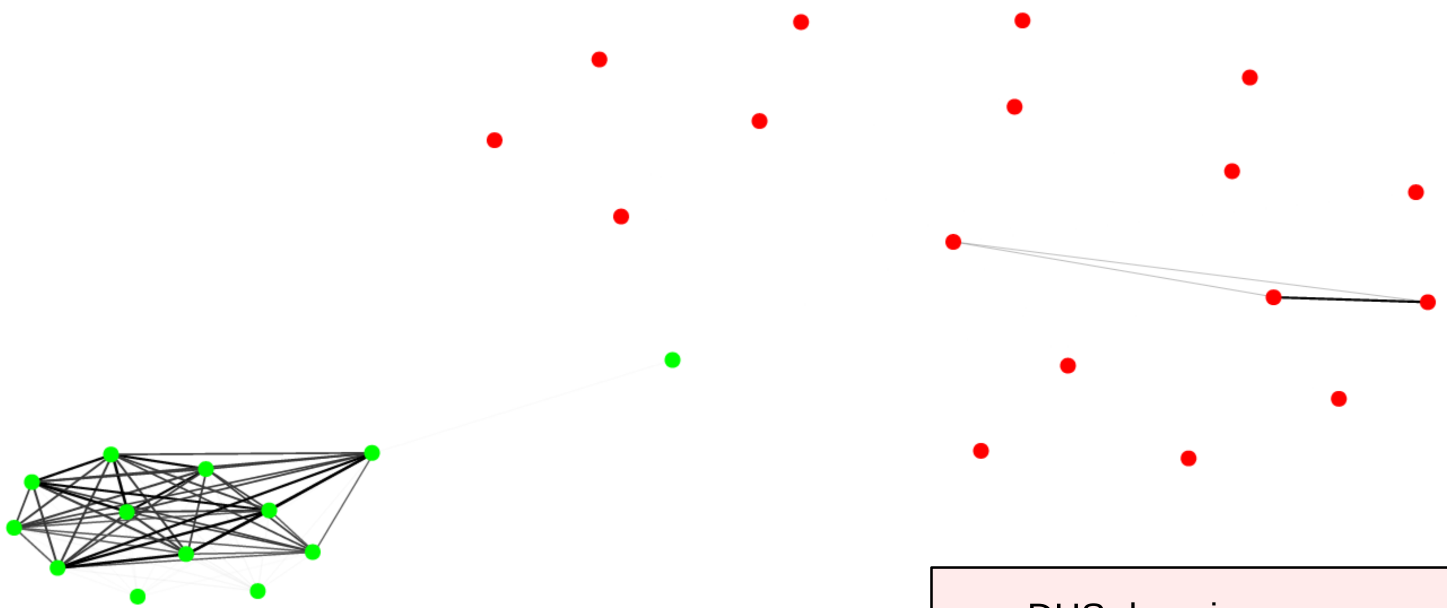
Create a graph of indicators where the edge weight is determined by **Jaccard similarity**:

$$J(Ind_A, Ind_B) = \frac{|A \cap B|}{|A \cup B|}$$

Where A and B are the sets of tickets containing Indicator A and Indicator B respectively.

Community detection

Community detection algorithms find groups of vertices that are interconnected



- MD5
- 3 phishing email addresses
- Filename
- File paths
- IPs

- DHS domain
- Email for submitting virus information
- DHS informational website

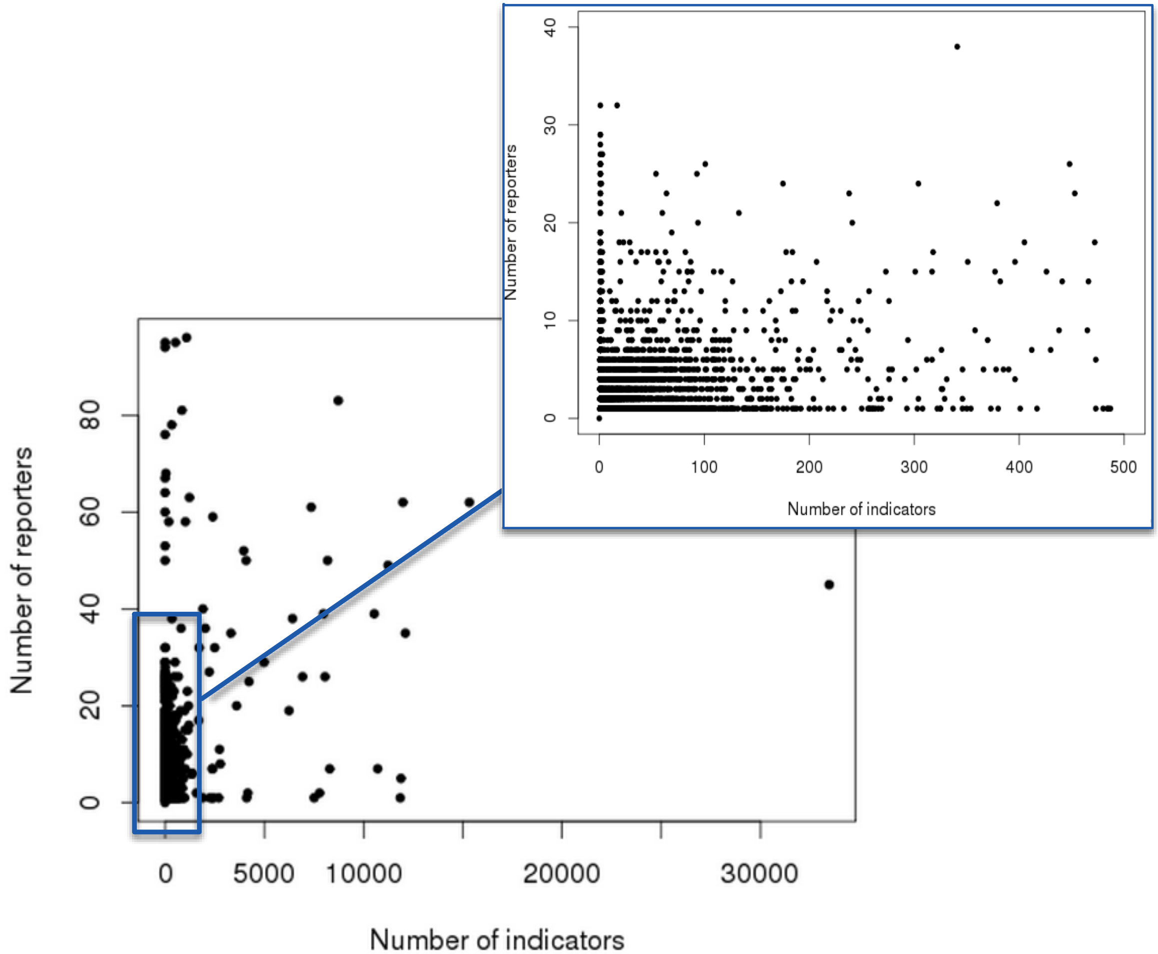
Communities as objects

group	email	filename	filepath	fqdn	ipv4addr	ipv6addr	md5	regkey	sha1	ssdeep	url	useragent	indcount	mindcount	sindcount	badcount	tcount	agcount
2	58	382	37	1150	8270	0	78	0	2	0	1248	8	11233	948	574	3237	891	49
844	687	760	145	3325	5588	2	575	0	13	3	348	533	11979	5402	2051	1076	8171	62
955	196	905	1393	686	6984	0	111	142	60	0	44	18	10539	1244	605	807	2407	39
1066	22378	501	397	7342	1753	20	663	1	86	18	275	19	33453	2318	1106	2988	2255	45
1177	3805	663	165	1456	2324	3	113	7	24	3	133	24	8720	2134	754	663	12663	83
1288	12	34	58	313	669	0	32	0	0	0	19	41	1178	452	304	206	372	20
1399	404	570	325	2503	3145	3	337	5	61	0	642	189	8184	2221	904	1200	2860	50
1510	93	326	567	925	1654	2	187	9	38	2	92	67	3962	1533	593	414	3003	52

Each detected community has measurable characteristics

- Connectivity
- Number of reporters, indicators, indicator types, tickets
- Date ranges

Can find communities with particular characteristics, or communities similar to a reference community.



Integrating additional information

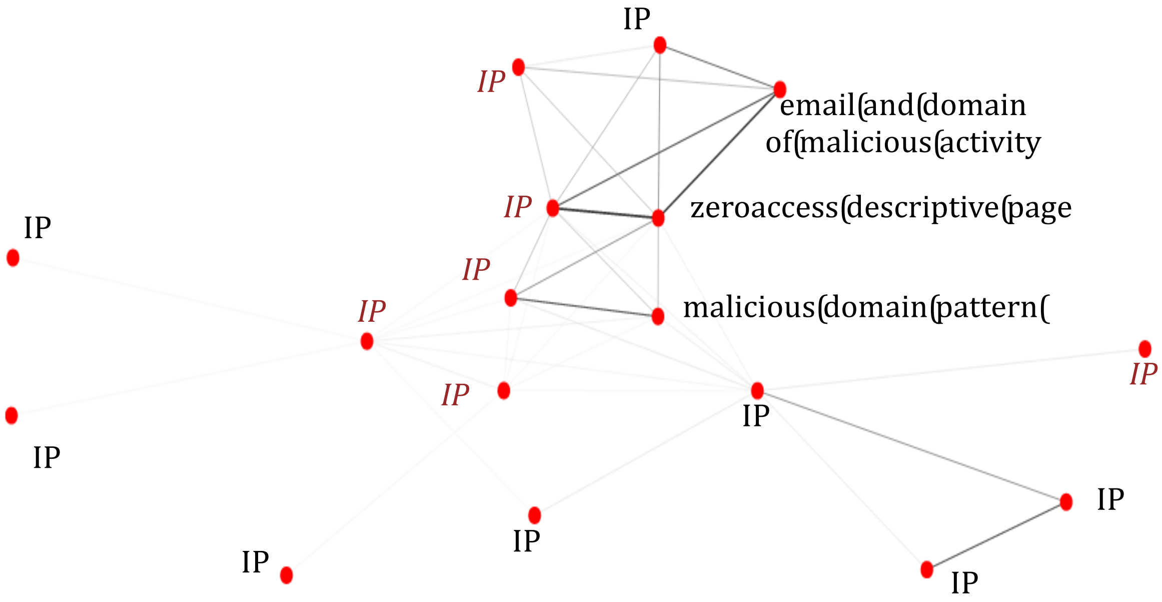
We matched indicators against blacklists.

(A) Can help interpret communities and sub-communities, or find interesting communities.

(B) Can supplement or correct blacklists.

Additional information could

- Supplement similarity metric
- Improve or tune community detection algorithm
- Tag or annotate communities



Continuing work

1. Find ‘interesting’ communities based on similarity to labeled examples.

2. Track evolution of a type of community over time. How do different types of communities develop?

3. Integrate expert information or additional data sources.

4. Explore value for predictive forecasting.

Summary

- We consider the tickets taken together as a sample of observations of coherent activities.
- We use statistical patterns in indicators across tickets and reporters to estimate similarity metrics and indicator communities.
- Communities can be more accessible, concise, and semantically coherent than large sets of individual indicators.
- This inferred structure can be integrated with additional information such as blacklists.
- Ongoing work will improve the integration of learned structure with additional information, forecasting, decision making

References

Network tie strength (similarity)

Gupte, M., & Eliassi-Rad, T. (2012). *Measuring tie strength in implicit social networks.* Proceedings of the 3rd Annual ACM Web Science Conference on - WebSci ’12, 109–118. doi:10.1145/2380718.2380734

Community detection algorithm InfoMap

M. Rosvall and C. T. Bergstrom, *Maps of information flow reveal community structure in complex networks*, PNAS 105, 1118 (2008)